Lecture 8: Pivoting rules

September 8, 2022 Kennesaw State University

1 A very degenerate problem

Here is the example we will consider today:

Problem 1. *Xerxes, Yvonne, and Zsuzsa decide to bake biscuits. However, before they begin baking, they start arguing about how they'll divide the biscuits:*

- *Xerxes says, "I'm not greedy; I just want at least one biscuit for every two biscuits that the two of you take."*
- *Yvonne says, "Well, last time we baked, Zsuzsa didn't do her fair share of the work! I want at least twice as many biscuits as she will get."*
- *Zsuzsa says, "Look, not all of us are master bakers. I'll do my best, and I feel like I deserve at least a quarter of the biscuits we make."*

What is the maximum number of biscuits that they can end up baking?

If Xerxes gets x biscuits, Yvonne gets y biscuits, and Zsuzsa gets z biscuits, then our inequalities are: $x \geq \frac{1}{2}(y+z)$, $y \geq 2z$, and $z \geq \frac{1}{4}(x+y+z)$. The bakers' joint goal is to maximize the total number $x+y+z$. Rearranging the inequalities, we can write the problem as:

$$\begin{array}{ll} \underset{x,y,z\in\mathbb{R}}{\text{maximize}} & x + y + z \\ \text{subject to} & -2x + y + z \leq 0 \\ & -y + 2z \leq 0 \\ & x + y - 3z \leq 0 \\ & x, y, z \geq 0. \end{array}$$

The astute observer will notice that (as usual with baking) if we find a feasible solution (x, y, z) then we can scale it up to $(2x, 2y, 2z)$ or even $(100x, 100y, 100z)$ without violating the inequalities. So it seems like there can't be any limit in the number of biscuits baked.

This is almost correct. The challenge here is to figure out if there's any division of biscuits that will make all three bakers happy. If not, then the only feasible solution is $(x, y, z) = (0, 0, 0)$ and no amount of scaling that up will get you biscuits.

If we add slack variables w_1, w_2, w_3 to the inequalities, then they get us an initial basic feasible solution; no two-phase simplex method needed here! Unfortunately, the initial dictionary we write down looks somewhat concerning...

[1] This document comes from the Math 3272 course webpage: https://facultyweb.kennesaw.edu/mlavrov/courses/3272-fall-2022.php

Here it is:

$$\begin{array}{rl}
\max \zeta = & 0 + x + y + z \\
\hline
w_1 = & 0 + 2x - y - z \\
w_2 = & 0 + y - 2z \\
w_3 = & 0 - x - y + 3z
\end{array}$$

Any of the three entering variables seem like equally good candidates. Let's just try making y the entering variable.

If we follow our usual procedure, then only w_1 and w_3 make it onto our "shortlist" of leaving variables. (Even this step doesn't seem entirely justified! Usually, if a variable is not on our shortlist, it's because pivoting on it is guaranteed to produce an infeasible dictionary. However, in this case, all three leaving variables will produce feasible dictionaries when we pivot, because we won't be able to leave point $(0, 0, 0)$ after this step.) The ratios for w_1 and w_3 are both $\frac{0}{1}$, meaning that we can't increase y past 0 before either of them becomes negative. This is a tie, so we can't tell which variable to pick; let's arbitrarily take w_1.

After solving $w_1 = 2x - y - z$ to get $y = 2x - w_1 - z$ and substituting this for y in the other equations, we get the dictionary below:

$$\begin{array}{rl}
\max \zeta = & 0 + 3x - w_1 \\
\hline
y = & 0 + 2x - w_1 - z \\
w_2 = & 0 + 2x - w_1 - 3z \\
w_3 = & 0 - 3x + w_1 + 4z
\end{array}$$

It sure doesn't seem like we're making any progress. However, there is still a positive reduced cost, so we can still keep going by pivoting on x.

Altogether, there are $\binom{6}{3} = 20$ ways to choose three basic variables in this problem. One of them turns out not to work: if you try to solve for x, w_1, and w_3, you end up having to take the inverse of a singular matrix. That leaves 19 feasible dictionaries, all of which describe the point $(x, y, z) = (0, 0, 0)$ in various ways.

What's even the point of pivoting, then?[2] Actually, there are two possible outcomes that would solve the problem for us:

- Suppose that one of these 19 dictionaries has all negative reduced costs. Then that formula for ζ proves that whenever $x_1, x_2, x_3, w_1, w_2, w_3 \geq 0$, we have $\zeta \leq 0$. In that case, we'd be able to conclude that $(0, 0, 0)$ is the only feasible solution.
- Suppose that one of these 19 dictionaries has an entering variable, with positive reduced cost, such that all the coefficients in that column are positive. Then we'd have a way to escape to infinity: by increasing that variable and keeping the other nonbasic variables at 0, we increase all the basic variables (and ζ) and discover that the linear program is unbounded.

The problem is that because of all the degenerate pivots we're doing, we can never tell if we're making progress toward either of these goals. In fact, we don't even have a clear proof that either of these outcomes is guaranteed to happen!

[2] Well, the point is $(0, 0, 0)$, of course. Haha.

2 Pivoting rules

A **pivoting rule** is a rule for making decisions in the simplex method in cases where our usual rules don't fully determine what to do. There are two situations in which we currently need the help of a pivoting rule:

1. If two or more nonbasic variables have a reduced cost with the correct sign (positive when maximizing, and negative when minimizing), then we don't know how to choose between them.

2. If two or more potential leaving variables are tied with an equal ratio, then we can bring either one of them out of the basis and get a feasible dictionary. Again, we don't know which one to choose.

We'd like to settle these two scenarios in a way which avoids **cycling**: going between the same set of dictionaries forever. Our secondary goal is to make decisions that speed up the simplex method.

For example, it seems like a generally useful heuristic to use the **highest-cost pivoting rule** to settle situation 1: when maximizing, pick the entering variable with the largest positive cost, and when minimizing, pick the entering variable with the most negative cost. The intuition is that this picks the direction in which the objective value improves as rapidly as possible.

On the other hand, in situation 2, it's not clear what to do. But if we're describing an algorithm completely, we need to specify how to break ties! One possibility is to go in order: write down a fixed ordered list of all our variables (such as (x, y, z, w_1, w_2, w_3)) and always pick the first variable on that list if there's a tie for the leaving variable.

Unfortunately, the combination of these two rules is not a winner. It is possible to come with examples in which it will cycle forever between different representations of the same corner point. (One example is given in section 3.2 of Vanderbei's textbook.)

It turns out that one good answer is to use **Bland's rule**. Here, once again, we pick a fixed ordered list of our variables. This time, we use that list to make decisions in both situation 1 and situation 2.

Fact 1. *Bland's rule prevents cycling: it can never return to a dictionary it's previously considered.*

I am calling this a "fact" and not a "theorem" because we will not prove it.

The drawback of Bland's rule is that it's slow: even though it never returns to the same feasible dictionary twice in degenerate cases, it tends to perform badly in cases with no degeneracy. That is, it often picks longer paths from the initial corner point to the optimal one. Intuitively, the reason this happens is that variables earlier in our list are both more likely candidates to enter the basis *and* more likely candidates to leave: so they end up flipping back and forth often. (Unfortunately, this property also plays a key role in the proof that Bland's rule prevents cycling.)

We'd like to come up with a rule that avoids cycling just by addressing situation 2 (how to choose leaving variables). That way, we can pair it with the highest-cost pivoting rule, which only addresses situation 1 (how to choose entering variables). The highest-cost pivoting rule is not the smartest rule there is, but it's good enough in many cases.

3 Lexicographic pivoting

The solution we're looking for is called **lexicographic pivoting**. To explain this rule, we'll begin with a different rule that's bad in many ways, but will provide useful intuition.

3.1 Intuition: random perturbations

Cycling can only happen when we have a degenerate pivoting step: otherwise, we're improving the objective value with every pivot, and can never return to a previous (worse) dictionary. Degenerate pivoting steps only happen when we have too many variables simultaneously equal to 0 at the same corner point.

In a randomly-chosen problem, this would never happen; once a corner point in $\mathbb{R}^n$ is determined as the intersection of n hyperplanes, another random hyperplane is very unlikely to pass exactly through that point. (In fact, in a formal way of defining that probability, the probability is 0.) Unfortunately, we don't usually solve randomly-chosen problems: our example today ends up with lots of degenerate pivots, even though we didn't do anything *that* weird.

But imagine if we took our system $A\mathbf{x} = \mathbf{b}$ and randomly adjusted the constants $\mathbf{b}$ by a small amount. For example, maybe we randomly adjust our initial dictionary as follows:

$$\begin{array}{rl} \max \zeta = & 0 + \ \ x + y + \ \ z \\ \hline w_1 = & 0 + 2x - y - \ \ z \\ w_2 = & 0 \qquad\quad + y - 2z \\ w_3 = & 0 - \ \ x - y + 3z \end{array} \quad \rightsquigarrow \quad \begin{array}{rl} \max \zeta = & \qquad\qquad\quad 0 + \ \ x + y + \ \ z \\ \hline w_1 = & 0.000878996 + 2x - y - \ \ z \\ w_2 = & 0.000534988 \qquad\quad + y - 2z \\ w_3 = & 0.000657869 - \ \ x - y + 3z \end{array}$$

Geometrically, we've taken each equation and pushed it by a random tiny amount. It is very unlikely that the result has even a single degenerate dictionary. So with this adjustment, none of our pivot steps will be degenerate, and so we'll never cycle. Of course, we're solving a slightly different problem now, but as long as our random adjustments were sufficiently small, our final answer will be very very very close to the answer to our original problem.

(Once we're done, we may even be able to recover the exact answer to the original problem, by assuming that our random adjustment doesn't change the optimal choice of basic variables.)

3.2 Actual lexicographic pivoting

The method of random perturbations works, but it's not very elegant: the solution we get is a tiny amount off from correct, the calculations become much messier, and it's hard to be certain what the threshold for "tiny adjustment" is before we end up solving a completely different problem.

The lexicographic pivoting rule is inspired by random perturbations in a "let's not, and say we did" kind of way. Instead of adding actual tiny random numbers to our constraints, we add variables $\epsilon_1, \epsilon_2, \epsilon_3, \dots, \epsilon_m$ that represent those tiny adjustments in symbolic form:

$$\begin{array}{rl} \max \zeta = & 0 + \ \ x + y + \ \ z \\ \hline w_1 = & 0 + 2x - y - \ \ z \\ w_2 = & 0 \qquad\quad + y - 2z \\ w_3 = & 0 - \ \ x - y + 3z \end{array} \quad \rightsquigarrow \quad \begin{array}{rl} \max \zeta = & 0 + \ \ x + y + \ \ z \\ \hline w_1 = & \epsilon_1 + 2x - y - \ \ z \\ w_2 = & \epsilon_2 \qquad\quad + y - 2z \\ w_3 = & \epsilon_3 - \ \ x - y + 3z \end{array}$$

The rule for dealing with these ϵ_i's is summarized by the inequality

$$1 \gg \epsilon_1 \gg \epsilon_2 \gg \cdots \gg \epsilon_m > 0.$$

What does this mean? Let's break it down step-by-step:

- Because $1 \gg \epsilon_1$, we say that in any comparison between an actual constant and ϵ_1 (or any other ϵ_i), the constant wins. For example, we would treat even 1.001 as bigger than $1 + 1\,000\,000\epsilon_1$. The ϵ_i's only help us break ties between the actual constants we've got.

  This makes sure that every pivoting step we do is also a valid pivoting step for the original problem. At the end, we'll be able to take the solution we got, drop all the ϵ_i's from it, and get a solution to the problem we wanted to solve.

- Similarly, $\epsilon_1 \gg \epsilon_2 \gg \epsilon_3 \gg \cdots \gg \epsilon_m$ say that in any comparison between two different constants ϵ_i and ϵ_j, the constant with the smaller subscript wins.

  This makes sure that after we add the ϵ_i's, we can never obtain a tie. Two expressions with ϵ_i's in them can only be equal if each of $\epsilon_1, \epsilon_2, \ldots, \epsilon_m$ has the same coefficient. However (more on this later) the coefficients of $\epsilon_1, \epsilon_2, \ldots, \epsilon_m$ track how we obtained each equation from our starting equations: so if two equations had the same coefficient on every ϵ_i, they'd actually be the same equation.

The lexicographic pivoting rule gets its name from this ordering.

3.3 Working through an example

Starting from the dictionary we had just now (which is repeated below on the left), let's pivot with y as our entering variable, as before. Now w_1 and w_3 are on the shortlist, but there's no longer a tie between them: the ratio $\frac{\epsilon_3}{1}$ is smaller than $\frac{\epsilon_1}{1}$, so w_3 is the *only* possibly leaving variable. After solving its equation for y, we get $y = \epsilon_3 - x + 3z - w_3$, which we then substitute for x in our other equations. The result is shown on the right:

$$\begin{array}{rl} \max \zeta = & 0 + x + y + z \\ \hline w_1 = & \epsilon_1 + 2x - y - z \\ w_2 = & \epsilon_2 + y - 2z \\ w_3 = & \epsilon_3 - x - y + 3z \end{array} \quad \rightsquigarrow \quad \begin{array}{rl} \max \zeta = & \epsilon_3 + 4z - w_3 \\ \hline w_1 = & (\epsilon_1 - \epsilon_3) + 3x - 4z + w_3 \\ w_2 = & (\epsilon_2 + \epsilon_3) - x + z - w_3 \\ y = & \epsilon_3 - x + 3z - w_3 \end{array}$$

We've made an infinitesimal amount of progress: the objective value has improved from 0 to ϵ_3. (Granted, that's pretty much the least amount of progress possible, but so what.) Note that all three basic variables are still positive: in particular, $\epsilon_1 - \epsilon_3 > 0$.

There is only one positive reduced cost: it is on z. No need to compare ratios: the only possible leaving variable when z enters the basis is w_1. The resulting dictionary is

$$\begin{array}{rl} \max \zeta = & \epsilon_1 + 3x - w_1 \\ \hline z = & (\frac{1}{4}\epsilon_1 - \frac{1}{4}\epsilon_3) + \frac{3}{4}x - \frac{1}{4}w_1 + \frac{1}{4}w_3 \\ w_2 = & (\frac{1}{4}\epsilon_1 + \epsilon_2 + \frac{3}{4}\epsilon_3) - \frac{1}{4}x - \frac{1}{4}w_1 - \frac{3}{4}w_3 \\ y = & (\frac{3}{4}\epsilon_1 + \frac{1}{4}\epsilon_3) + \frac{5}{4}x - \frac{3}{4}w_1 - \frac{1}{4}w_3 \end{array}$$

Now the only positive reduced cost is on x. Once again, there is only one possible leaving variable, which is w_2. After a third pivot step, we get:

$$\begin{array}{rl}
\max \zeta = & (4\epsilon_1 + 12\epsilon_2 + 9\epsilon_3) - 4w_1 - 12w_2 - 9w_3 \\
\hline
z = & (\epsilon_1 + 3\epsilon_2 + 2\epsilon_3) - w_1 - 3w_2 - 2w_3 \\
x = & (\epsilon_1 + 4\epsilon_2 + 3\epsilon_3) - w_1 - 4w_2 - 3w_3 \\
y = & (2\epsilon_1 + 5\epsilon_2 + 4\epsilon_3) - 2w_1 - 5w_2 - 4w_3
\end{array}$$

Since the reduced costs of w_1, w_2, w_3 are all negative, this tells us that we've "maximized" ζ: $4\epsilon_1 + 12\epsilon_2 + 9\epsilon_3$ is the highest possible value it could have. Of course, just like every other value we saw for ζ, it rounds to 0. To get our final answer, we set $\epsilon_1 = \epsilon_2 = \epsilon_3 = 0$ and get that $(0, 0, 0)$ really is our optimal solution.

3.4 Shortcuts (optional)

If you look back at our work, and especially at the final dictionary, you may see a pattern: the coefficients on $\epsilon_1, \epsilon_2, \epsilon_3$ end up matching the coefficients on w_1, w_2, w_3, up to a sign.

This is not a coincidence. When we start out writing our first dictionary, we have equations $w_1 = \epsilon_1 + \cdots$, $w_2 = \epsilon_2 + \cdots$, and $w_3 = \epsilon_3 + \cdots$. After that, the cardinal rule of working with equations is that we always do the same thing to both sides. So it will always be true that:

- When w_i and ϵ_i are on opposite sides of the equation, they have the same coefficient.
- When w_i and ϵ_i are on the same side of the equation, they have opposite coefficients (same magnitude, but different sign).
- When w_i doesn't appear in an equation, neither does ϵ_i.

This means that in principle, we can use the lexicographic pivoting rule without actually writing down the ϵ_i's. As long as there's no ties between potential leaving variables, it's business as usual. Once there's a tie, use these rules to figure out the coefficients of $\epsilon_1, \epsilon_2, \ldots$ and break the tie.

The variables we use for this are w_1, w_2, w_3 in this problem because those were the basic variables in our initial dictionary, which we adjusted by adding $\epsilon_1, \epsilon_2, \epsilon_3$. In general, whatever our initial basic variables are, those will be the variables we can use to deduce the coefficients of the ϵ_i's.